



Communication Scheduling for Time-Triggered Systems

Pop, Paul; Eles, Petru; Peng, Zebo

Published in:
Euromicro Conference on Real-Time Systems. Proceedings

Publication date:
1999

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Pop, P., Eles, P., & Peng, Z. (1999). Communication Scheduling for Time-Triggered Systems. *Euromicro Conference on Real-Time Systems. Proceedings*.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

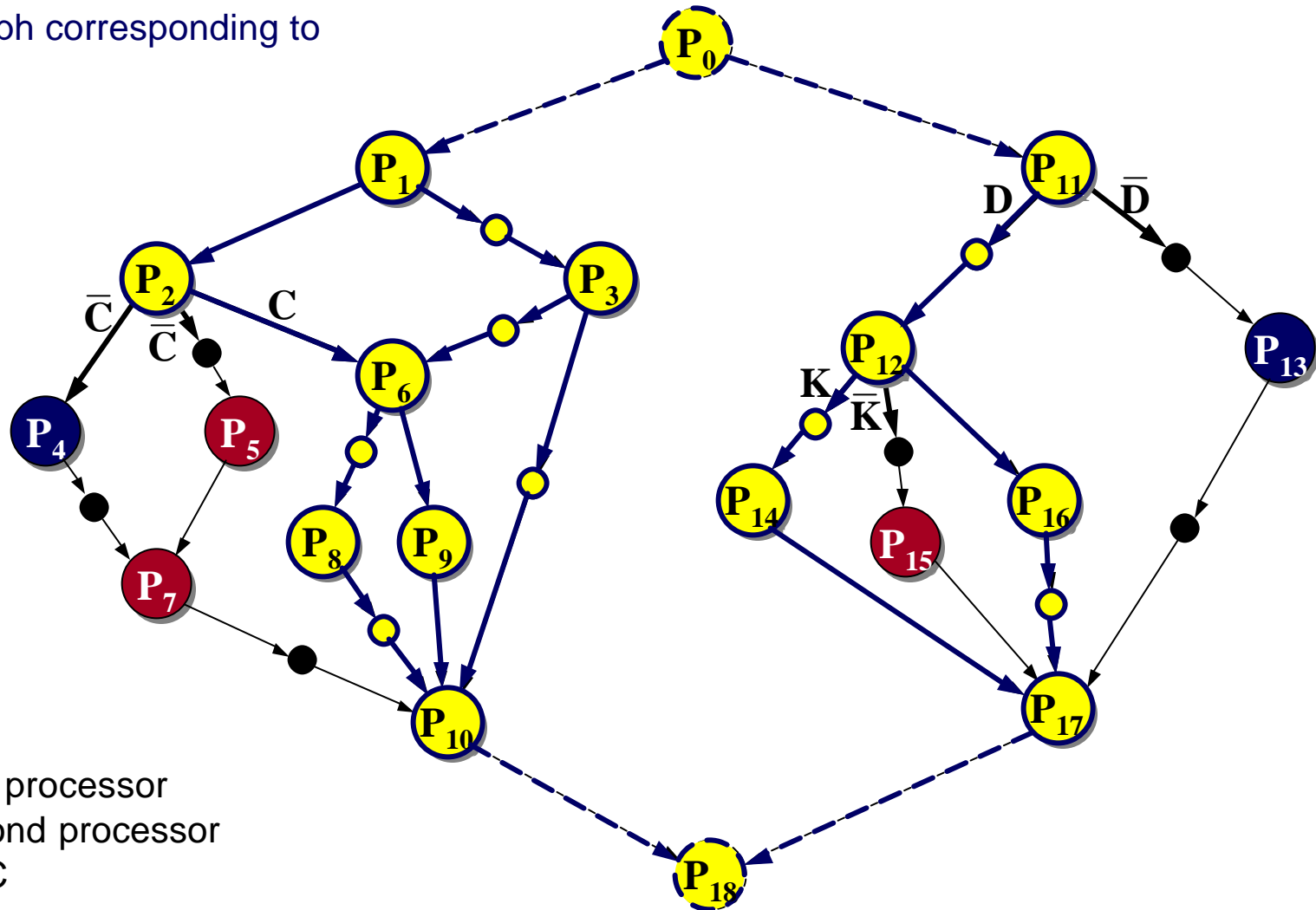
Communication Scheduling for Time-Triggered Systems

Paul Pop, Petru Eles and Zebo Peng

**Dept. of Computer and Information Science
Linköping University
Sweden**

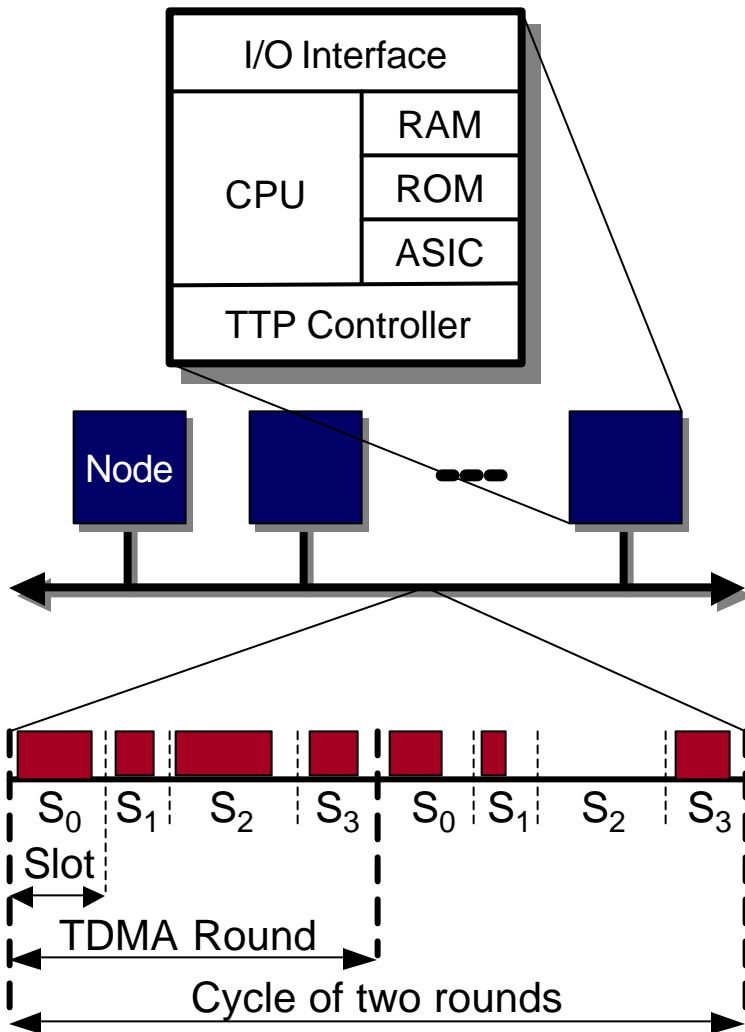
Conditional Process Graph

Subgraph corresponding to $D \wedge C \wedge K$



- First processor
- Second processor
- ASIC

Hardware Architecture



- Safety-critical distributed embedded systems.
- Nodes connected by a broadcast communication channel.
- Nodes consisting of: TTP controller, CPU, RAM, ROM, I/O interface, (maybe) ASIC.
- Communication between nodes is based on the time-triggered protocol.
- Buss access scheme: time-division multiple-access (TDMA).
- Schedule table located in each TTP controller: message descriptor list (MEDL).

Problem Formulation

Input

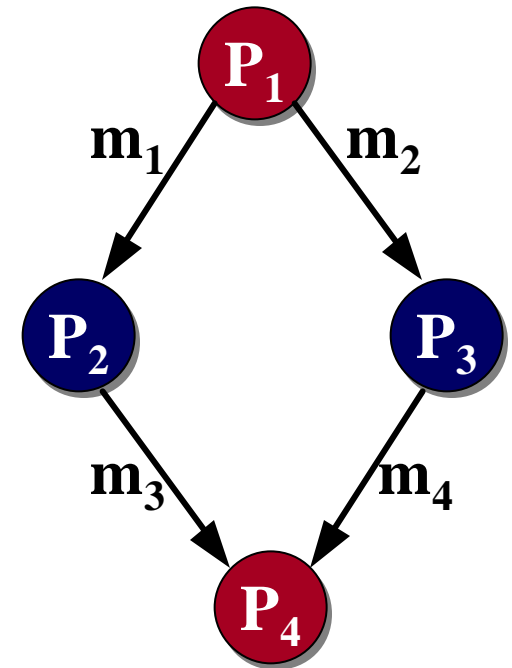
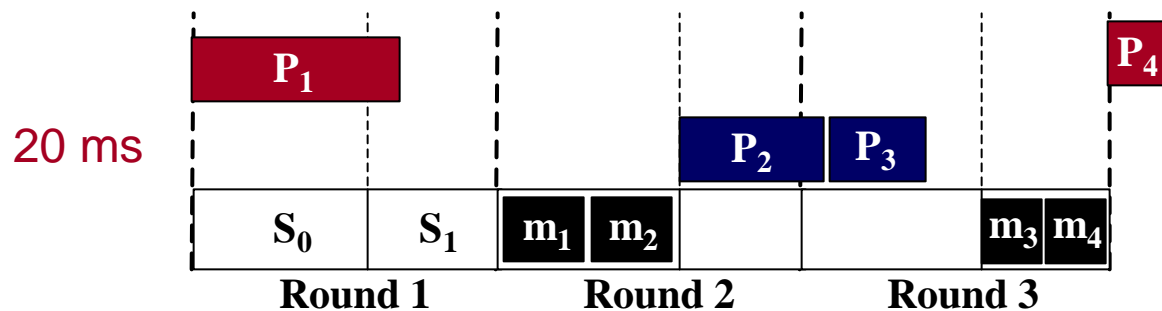
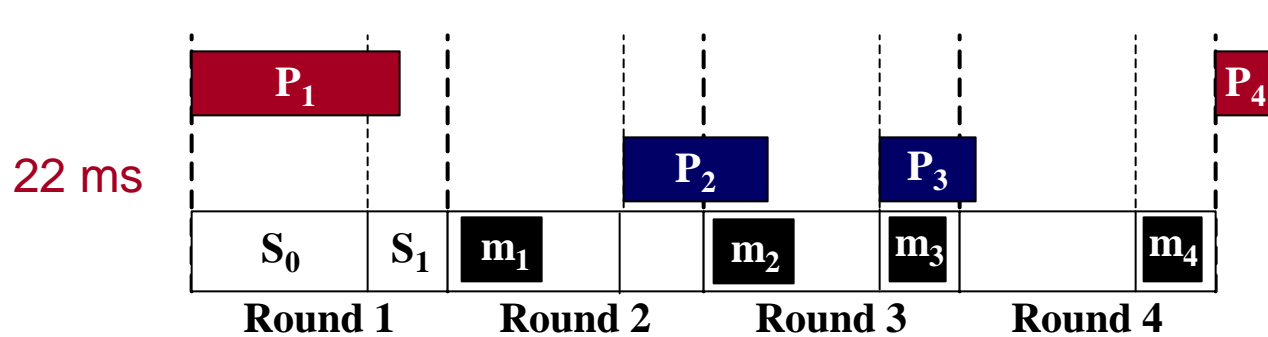
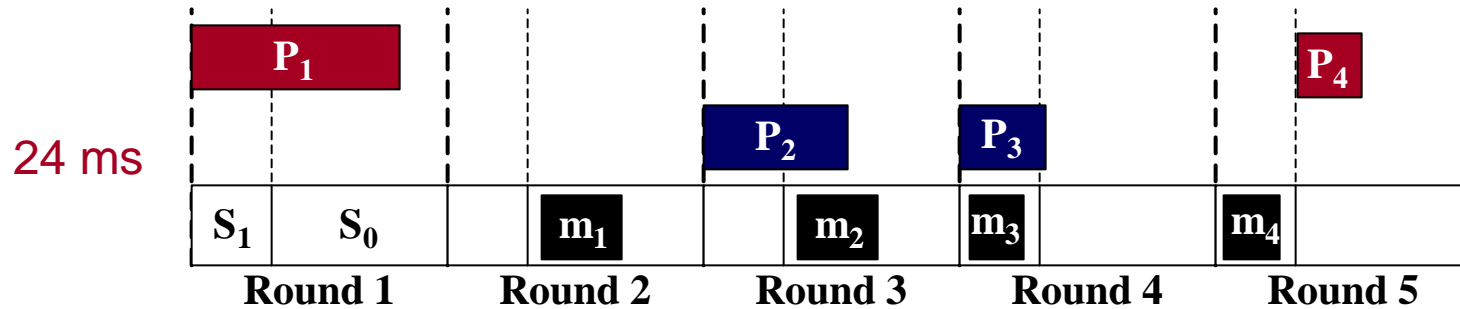
- Safety-critical application with several operating modes.
- Each operating mode is modelled by a conditional process graph.
- The system architecture and mapping of processes to nodes are given.
- The worst case delay of a process is known:

$$T_{P_i} = (d_{PA} + t_{P_i} + q_{C_1} + q_{C_2})$$
$$q_{C_1} = \sum_{i=1}^{N_{out}^{local}(P_i)} d_{S_i} \quad q_{C_2} = \sum_{i=1}^{N_{out}^{remote}(P_i)} d_{KS_i} + \sum_{i=1}^{N_{in}^{remote}(P_i)} d_{KR_i}$$

Output

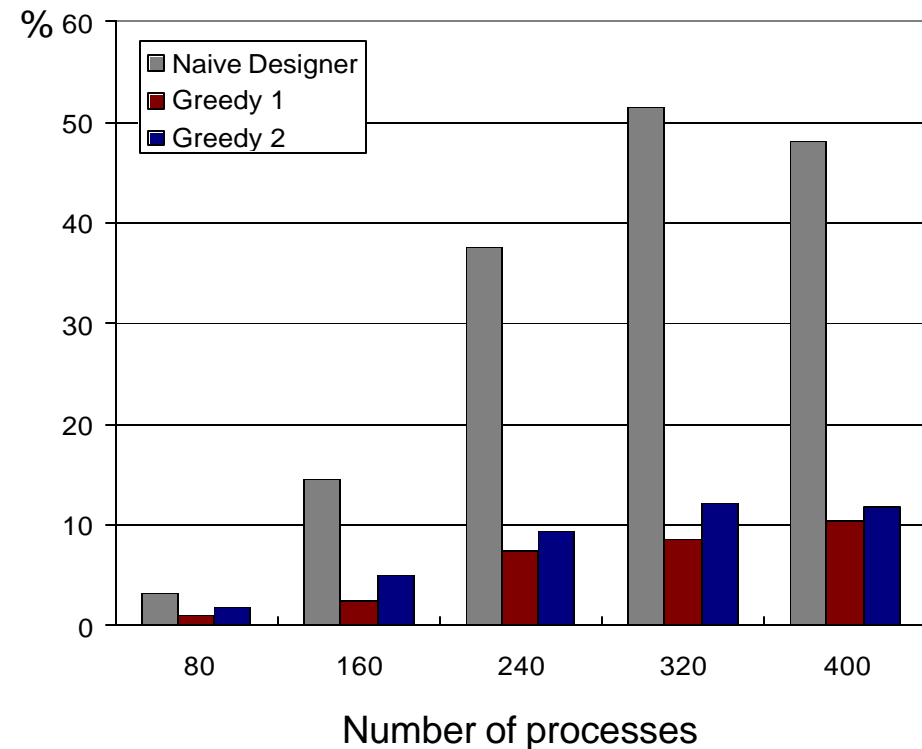
- Local schedule tables for each node and the MEDL for the TTP controllers.
- Delay on the system execution time for each operating mode, so that this delay is as small as possible.

Scheduling Example



Experimental Results

Average percentage deviations
from the lengths of near-optimal schedules



- The Greedy Approach is producing accurate results in a very short time (few seconds for graphs with 400 processes).
- Greedy 1 performs slightly better than Greedy 2, but it is a bit slower.
- SA finds near-optimal results in a reasonable time (few minutes for graphs with 80 processes and 275 minutes for graphs with 400 processes).
- A real-life example implementing a vehicle cruise controller validated our approach.